

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of:

Ryuta TANAKA et al.

Application No.: Unassigned

Group Art Unit: Unassigned

Filed: November 16, 2001

Examiner:

For: MULTIPROCESSOR SYSTEM, MULTIPROCESSOR CONTROL METHOD, AND  
MULTIPROCESSOR CONTROL PROGRAM RETAINING COMPUTER-READABLE  
RECORDING MEDIUM



**SUBMISSION OF CERTIFIED COPY OF PRIOR FOREIGN  
APPLICATION IN ACCORDANCE  
WITH THE REQUIREMENTS OF 37 C.F.R. § 1.55**

Assistant Commissioner for Patents  
Washington, D.C. 20231

Sir:

In accordance with the provisions of 37 C.F.R. § 1.55, the applicant(s) submit(s)  
herewith a certified copy of the following foreign application:

Japanese Patent Application No. 2001-097062

Filed: March 29, 2001

It is respectfully requested that the applicant(s) be given the benefit of the foreign filing  
date(s) as evidenced by the certified papers attached hereto, in accordance with the  
requirements of 35 U.S.C. § 119.

Date: November 16, 2001

By: \_\_\_\_\_

Respectfully submitted,  
STAAS & HALSEY LLP

James D. Halsey, Jr.  
Registration No. 22,729

700 11th Street, N.W., Ste. 500  
Washington, D.C. 20001  
(202) 434-1500

日 本 国 特 許 庁  
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出 願 年 月 日  
Date of Application:

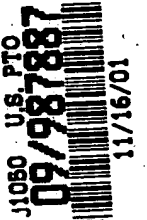
2001年 3月29日

出 願 番 号  
Application Number:

特願2001-097062

出 願 人  
Applicant(s):

富士通株式会社

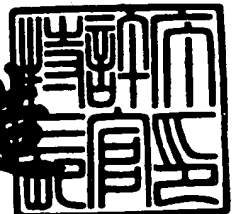


CERTIFIED COPY OF  
PRIORITY DOCUMENT

2001年 8月10日

特許庁長官  
Commissioner,  
Japan Patent Office

及川耕造



出証番号 出証特2001-3072252

【書類名】 特許願

【整理番号】 01-50182

【提出日】 平成13年 3月29日

【あて先】 特許庁長官殿

【国際特許分類】 G06F 15/16

【発明の名称】 マルチプロセッサシステム, マルチプロセッサ制御方法  
, マルチプロセッサ制御プログラムおよび同プログラム  
を記録したコンピュータ読取可能な記録媒体

【請求項の数】 5

【発明者】

【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士通  
株式会社内

【氏名】 田中 竜太

【発明者】

【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士通  
株式会社内

【氏名】 熊本 乃親

【発明者】

【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士通  
株式会社内

【氏名】 鶴田 徹

【発明者】

【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士通  
株式会社内

【氏名】 田中 律子

【発明者】

【住所又は居所】 福岡県福岡市博多区博多駅前三丁目2番8号 富士通  
九州ディジタル・テクノロジー株式会社内

【氏名】 岩崎 尚之

【発明者】

【住所又は居所】 神奈川県川崎市中原区上小田中4丁目1番1号 富士通  
株式会社内

【氏名】 石原 輝雄

【特許出願人】

【識別番号】 000005223

【氏名又は名称】 富士通株式会社

【代理人】

【識別番号】 100092978

【弁理士】

【氏名又は名称】 真田 有

【電話番号】 0422-21-4222

【手数料の表示】

【予納台帳番号】 007696

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9704824

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 マルチプロセッサシステム、マルチプロセッサ制御方法、マルチプロセッサ制御プログラムおよび同プログラムを記録したコンピュータ読取可能な記録媒体

【特許請求の範囲】

【請求項 1】 2 以上のプロセッサエレメントと、

前記 2 以上のプロセッサエレメントの内から、一つのプロセッサエレメントを被選択プロセッサエレメントとして選択して動作させる制御部と、

該制御部による前記被選択プロセッサエレメントの切替時に、切替前の被選択プロセッサエレメントから切替後の被選択プロセッサエレメントへ受け渡すべき情報を格納する記憶部とをそなえたことを特徴とする、マルチプロセッサシステム。

【請求項 2】 該制御部が、前記被選択プロセッサエレメントの切替時に、前記情報を前記切替前の被選択プロセッサエレメントから該記憶部に格納させ、前記情報の格納後に、前記切替前の被選択プロセッサエレメントを停止させるとともに、該記憶部に格納された前記情報を用いて、前記切替後の被選択プロセッサエレメントを動作させることを特徴とする、請求項 1 記載のマルチプロセッサシステム。

【請求項 3】 2 以上のプロセッサエレメントの内から一つのプロセッサエレメントを被選択プロセッサエレメントとして選択して動作させるように制御するマルチプロセッサ制御方法であって、

前記被選択プロセッサエレメントの切替時に、切替前の被選択プロセッサエレメントから切替後の被選択プロセッサエレメントへ受け渡すべき情報を、前記切替前の被選択プロセッサエレメントから記憶部に格納する格納ステップと、

前記情報の格納後に、該切替前の被選択プロセッサエレメントを停止させるとともに、前記記憶部に格納された前記情報を用いて、切替後の被選択プロセッサエレメントを動作させる切替ステップとをそなえたことを特徴とする、マルチプロセッサ制御方法。

【請求項 4】 2 以上のプロセッサエレメントの内から一つのプロセッサエ

レメントを被選択プロセッサエレメントとして選択して動作させるように制御するマルチプロセッサ制御プログラムであって、

前記被選択プロセッサエレメントの切替時に、切替前の被選択プロセッサエレメントから切替後の被選択プロセッサエレメントへ受け渡すべき情報を、前記切替前の被選択プロセッサエレメントから記憶部に格納する格納ステップと、

前記情報の格納後に、該切替前の被選択プロセッサエレメントを停止させるとともに、前記記憶部に格納された前記情報を用いて、切替後の被選択プロセッサエレメントを動作させる切替ステップとを、コンピュータに実行させることを特徴とする、マルチプロセッサ制御プログラム。

【請求項 5】 2 以上のプロセッサエレメントの内から一つのプロセッサエレメントを被選択プロセッサエレメントとして選択して動作させるように制御するマルチプロセッサ制御プログラムを記録したコンピュータ読取可能な記録媒体であって、

該マルチプロセッサ制御プログラムが、

前記被選択プロセッサエレメントの切替時に、切替前の被選択プロセッサエレメントから切替後の被選択プロセッサエレメントへ受け渡すべき情報を、前記切替前の被選択プロセッサエレメントから記憶部に格納する格納ステップと、

前記情報の格納後に、該切替前の被選択プロセッサエレメントを停止させるとともに、前記記憶部に格納された前記情報を用いて、切替後の被選択プロセッサエレメントを動作させる切替ステップとを、該コンピュータに実行させることを特徴とする、マルチプロセッサ制御プログラムを記録したコンピュータ読取可能な記録媒体。

【発明の詳細な説明】

【0 0 0 1】

【発明の属する技術分野】

本発明は、2 以上のプロセッサエレメントをそなえるマルチプロセッサシステムに関し、例えば、携帯電話端末や P H S (Personal Handyphone System) 端末等の携帯型の情報通信端末装置に用いて好適な、マルチプロセッサシステム、マルチプロセッサ制御方法、マルチプロセッサ制御プログラムおよび同プロ

グラムを記録したコンピュータ読取可能な記録媒体に関する。

【0002】

【従来の技術】

例えば、携帯電話端末や、PHS (Personal Handyphone System) 端末等の携帯型の情報通信端末装置においては、OS (Operating System) 上で行なわれる各種の制御処理の他、音声や画像、動画の圧縮／伸長等の処理や入出力信号を取り扱う処理等が行なわれている。一般に、OS上で行なわれる各種の処理にはMPU (Micro Processing Unit) が適しており、一方、音声や画像、動画を取り扱う処理や入出力信号を取り扱う処理等のリアルタイム性が要求される処理には、DSP (Digital Signal Processor) が適している。

【0003】

そして、情報処理装置において、これらの異なる処理を行なうためのDSPマイコンやマルチプロセッサシステムが知られている。

DSPマイコンは、OS処理を効率的に処理することができるMPU機能と、信号処理等を高速に処理することができるDSP機能とを統合したプロセッサアーキテクチャによって構成されたものである。そして、一つのプログラムにおいて、用途に応じてMPU命令とDSP命令とを混在させて記述することにより、このDSPマイコンに統合されたMPU機能とDSP機能とを使い分けることができる。

【0004】

また、このプログラミングに際して、既存のMPUと互換性があるMPU命令を使用することにより既存のOSをDSPマイコン上に移植することが比較的容易になり、又、既存のDSPと互換性があるDSP命令を使用することにより、既存のDSP用信号処理ソフト資産を活用することができ、更に、OS管理下のタスクとしてDSP命令を記述する事もできる。

【0005】

さらに、DSPマイコンのアーキテクチャの設計に際して、MPU機能もしくはDSP機能のいずれか一方の特性に合わせてパイプライン構造を成すように構成することによって、より高速な処理を実現することができる。

例えば、MPU機能としての特性に合わせて、様々の制御条件に応じた分岐オーバーヘッドを減らす工夫を行ないながら多段パイプライン化を図ることにより、その処理を高速化することができる。又、DSP機能としての特性に合わせて、メモリのロード／ストアのオーバーヘッドを削減するとともに、複数の積和演算をパイプライン化することによって時間軸方向に圧縮し処理を高速化することができ、又、並列処理を行なって空間方向へ展開し高速にループ処理を行なうことによって処理を高速化することができる。

## 【0006】

マルチプロセッサシステムは、複数のプロセッサエレメントをそなえて構成されたものであって、例えば、MPUとDSPとをそなえて構成することにより、MPUにOS上の処理を行なわせるとともに、DSPによる信号処理を制御することが可能となる。そして、従来のマルチプロセッサシステムにおいては、既存のMPUと既存のDSPとをそれぞれ用いて構成することにより、システムの構築を容易にすることができるとともに、既存ソフト資産を有効に利用することができる。

## 【0007】

## 【発明が解決しようとする課題】

しかしながら、DSPマイコンは、上述の如くMPUのパイプライン構造にDSP機能を統合することによって構成されているので、DSPマイコンにおいては必ずしもDSPにとって最適なパイプライン構成が形成されるものではなく、これにより、DSPの性能の低下が生じるという課題がある。すなわち、DSPマイコンにおいては、MPUとDSPとの双方の性能を100%発揮できるように設計することが困難である。

## 【0008】

また、DSPマイコンのLSI開発においては、DSPマイコンのための新しいアーキテクチャ開発が必要であり、かかる開発のために莫大な期間や費用が必要であるという課題もある。

特に、DSPマイコンは、MPUとDSPという本来アーキテクチャの異なるプロセッサパイプラインを統合して構成されているので、それぞれの処理に最適



なアーキテクチャ設計を市場ニーズのサイクルに合わせて短期開発していくことは困難であり、又、MPUもしくはDSPの性能を最大限に発揮することも困難である。更に、DSPマイコンと既存のMPUやDSPとの互換性を維持した設計を行なうことは、実際、困難であるので、既存のMPUやDSPの各ソフト資産の流用が困難であるという課題もある。

## 【0009】

一方、従来のマルチプロセッサシステムにおいては、互いに機能が異なるMPU命令とDSP命令とを効果的に混在させた利便性の高いプログラミングを行なうことは困難である。又、従来のマルチプロセッサシステムにおいては、各プロセッサエレメントがそれぞれ独自に動作するようになっているので、各プロセッサエレメントで実行されているプログラムの処理の流れを把握することが困難であり、プログラミングやデバッグが困難であるという課題もある。

## 【0010】

さらに、従来のマルチプロセッサシステムにおいては、MPUとDSPとの間でデータの受け渡しを行なうために排他制御処理や同期処理等を行なう必要があるので、これらの処理を考慮したソフトウェア開発に負担がかかるという課題があり、更に、これらの排他制御処理や同期処理におけるオーバーヘッドが性能低下をもたらすという課題もある。

## 【0011】

また、携帯電話等の情報処理端末においては、消費電力を低減することが要求されているが、従来のマルチプロセッサシステムにおいては、各プロセッサエレメントが同時に動作することにより、その消費電力を低減することが困難である。

また、従来のマルチプロセッサシステムにおいては、MPUとDSPとが疎結合によってデータを共有できるようになっているものの、あくまでMPUとDSPとが互いに独立するようにそれぞれを動作させるようになっているので、一つのプログラム中にMPU命令とDSP命令を混在させる事は困難である。

## 【0012】

さて、C言語などの高級言語を利用して記述された1本のプログラムを、コン

パイラ等のツールによって処理内容を判断し、特定の処理単位（関数単位、スレッド単位等）に分割した後に、これらの分割された処理をその処理内容に応じて最適な各プロセッサに割り当てて、同期をとりながらマルチプロセッサ上で分散処理する事が可能であれば、プログラマは、MPU命令での処理とDSP命令での処理とを効率よく使い分ける、見かけ上1本のプログラムを記述できることに等しい。

【0013】

ところが実際には、各処理単位をどのプロセッサで処理するのが最適であるかを、プログラミングツールがその処理内容に基づいて自動判定することは技術的に非常に困難であり、そのような最適化を完全自動で行なうツールは普及していない。又、従来のマルチプロセッサシステムで実現していた割込制御では、一本化されて記述されたプログラムが、複数のプロセッサ間で同期をとりながら1つのプログラムの流れとして動作していくためには、集中管理型の割り込みコントローラを用いても実現する事ができない。これは、プログラムの処理過程において、複数のプロセッサ間を跨いだ後に割り込みが発生する場合などに、1つのプログラムの流れが崩れて破綻してしまう為である。

【0014】

本発明は、このような課題に鑑み創案されたもので、2以上のプロセッサエレメントの各機能を最大限に生かすことができるとともに、これらのプロセッサエレメント間における情報の受け渡しを確実に行なうことができ、更に、消費電力を低減することができるようにした、マルチプロセッサシステム、マルチプロセッサ制御方法、マルチプロセッサ制御プログラムおよび同プログラムを記録したコンピュータ読取可能な記録媒体を提供することを目的とする。

【0015】

【課題を解決するための手段】

このため、本発明のマルチプロセッサシステム（請求項1）は、2以上のプロセッサエレメントと、これらの2以上のプロセッサエレメントの内から、一つのプロセッサエレメントを被選択プロセッサエレメントとして選択して動作させる制御部と、この制御部による被選択プロセッサエレメントの切替時に、切替前の

被選択プロセッサエレメントから切替後の被選択プロセッサエレメントへ受け渡すべき情報を格納する記憶部とをそなえたことを特徴としている。

## 【 0 0 1 6 】

また、本発明のマルチプロセッサ制御方法（請求項 3）は、2 以上のプロセッサエレメントの内から一つのプロセッサエレメントを被選択プロセッサエレメントとして選択して動作させるように制御するためのものであって、被選択プロセッサエレメントの切替時に、切替前の被選択プロセッサエレメントから切替後の被選択プロセッサエレメントへ受け渡すべき情報を、切替前の被選択プロセッサエレメントから記憶部に格納する格納ステップと、情報の格納後に、切替前の被選択プロセッサエレメントを停止させるとともに、記憶部に格納された情報を用いて、切替後の被選択プロセッサエレメントを動作させる切替ステップとをそなえたことを特徴としている。

## 【 0 0 1 7 】

さらに、本発明のマルチプロセッサ制御プログラム（請求項 4）は、上述した格納ステップおよび切替ステップをコンピュータに実行させるものであり、本発明のコンピュータ読取可能な記録媒体（請求項 5）は、上述したマルチプロセッサ制御プログラムを記録したものである。

これらにより、切替後の被選択プロセッサエレメントが、切替前の被選択プロセッサエレメントから切替後の被選択プロセッサエレメントへ受け渡すべき情報を用いて動作することができる。

## 【 0 0 1 8 】

なお、被選択プロセッサエレメントの切替時に、情報を切替前の被選択プロセッサエレメントから記憶部に格納させ、この情報の格納後に、切替前の被選択プロセッサエレメントを停止させるとともに、記憶部に格納された情報を用いて、切替後の被選択プロセッサエレメントを動作させてもよく、これにより、切替後に、切替後の被選択プロセッサエレメントだけを、記憶部に格納された情報を用いて動作させることができる（請求項 2）。

## 【 0 0 1 9 】

## 【発明の実施の形態】

以下、図面を参照して本発明の実施の形態を説明する。

(A) 第1実施形態の説明

図1および図2は本発明の第1実施形態としてのマルチプロセッサシステム1aの構成を説明するためのもので、図1はその機能構成を示すブロック図、図2は本第1実施形態のマルチプロセッサシステムにおける機能構成および信号の流れを説明するためのブロック図である。

【0020】

本第1実施形態のマルチプロセッサシステム1aは、携帯電話端末や、PHS (Personal Handyphone System) 端末等の携帯型の情報処理装置における各種の制御・処理を行なうものであって、図1に示すように、プロセッサエレメント (以下PEという) 2a, 2b, PE制御部 (制御部) 3および共有メモリ部 (記憶部) 4とをそなえて構成されている。

【0021】

PE2aおよびPE2bは、ともに情報処理装置における種々の処理を行なうプロセッサエレメント (被選択プロセッサエレメント) である。これらのPE2aとPE2bとは、互いに異なる機能を有するものであり、本実施形態においては、PE2aとして、例えば、OS (Operating System) 上で行なわれる各種の制御処理に好適なMPU (Micro Processing Unit) がそなえられており、又、PE2bとして、例えば、音声処理や画像処理、入出力処理等のリアルタイム性が求められる処理に好適なDSP (Digital Signal Processor) がそなえられている。

【0022】

これらのPE2aおよびPE2bは、既存のアーキテクチャコアを採用し、PE制御部3からのプロセッサ停止信号 (プロセッサエレメント用割込信号) による停止制御を行なうことができ、又、これらのPE2a及びPE2bからはPE制御部3に対する切替要求信号を出力できるようになっている。

なお、PE2aやPE2bからのPE制御部3に対する切替要求信号については、切替要求命令の発行によって出力されるような仕組みにより実現でき、これらの機能をそなえたプロセッサエレメントを開発することは既存のIP (Intell

ectual Property) コアを流用することにより容易に実現することができる。又、本第1実施形態においては、OSによるタスク管理などが特に行なわれない例について説明するものであり、PE2aとPE2bとにそれぞれ同等機能の割り込みハンドラを用意し、これらによる割込処理をそれぞれが行なうようになっている。

#### 【0023】

また、PE2aに対して要求された処理がPE2a以外のプロセッサエレメントによって行なわれるべきものである場合、すなわち、その処理がPE2bによって実行されるべきものである場合に、PE2aは、PE制御部3に対してプロセッサエレメントの切替要求信号を出力するようになっている。

同様に、PE2bに対して要求された処理がPE2b以外のプロセッサエレメントによって行なわれるべきものである場合、すなわち、その処理がPE2aによって実行されるべきものである場合に、PE2bは、PE制御部3に対してプロセッサエレメントの切替要求信号を出力するようになっている。

#### 【0024】

なお、以下、プロセッサエレメントを示す符号としては、複数のプロセッサエレメントのうち1つを特定する必要があるときには符号2aおよび2bを用いてPE2aおよびPE2bと示すが、任意のプロセッサエレメントを指すときには符号2を用いてプロセッサエレメント2と示す。

PE制御部3は、PE2aおよびPE2bの内から、一つのプロセッサエレメント2を被選択プロセッサエレメント2として選択して動作させるものであって、この被選択プロセッサエレメント2の切替時に、切替前の被選択プロセッサエレメント2から切替後の被選択プロセッサエレメント2へ受け渡すべき情報（以下、共有情報という）を切替前の被選択プロセッサエレメント2から後述する共有メモリ部4に格納させ、この共有情報の格納後に、切替前の被選択プロセッサエレメント2を停止させるとともに、共有メモリ部4に格納された共有情報を用いて、切替後の被選択プロセッサエレメント2を動作させるようになっている。

#### 【0025】

なお、上述した共有情報として、例えば、プログラムカウンタや関数の引数、

関数の戻り値およびスタックポインタ等が、切替前の被選択プロセッサエレメント 2 から切替後の被選択プロセッサエレメント 2 へ受け渡される。

また、図 1 に示すように、P E 制御部 3 には、情報処理装置における入力装置からの信号等の、P E 制御部 3 の外部からの信号（割込信号）が入力されるようになっており、P E 制御部 3 は、この割込信号に応じて、P E 2 a もしくは P E 2 b に対してプロセッサエレメント用割込信号を出力するようになっている。

【 0 0 2 6 】

P E 制御部 3 は、図 2 に示すように、割込管理部 3 1，割込管理テーブル 3 3 a，動作許可テーブル（テーブル） 3 2 および P E 実行許可部 3 4 をそなえて構成されている。なお、図 1 では、割込管理部 3 1，動作許可テーブル 3 2 および割込管理テーブル 3 3 a の図示は省略されている。又、図 2 では、共有メモリ部 4 の図示は省略されている。

【 0 0 2 7 】

図 3 は動作許可テーブル 3 2 の例を示す図、図 4 は割込管理テーブル 3 3 a の例を示す図である。動作許可テーブル 3 2 は、図 3 に示すように、P E 2 a もしくは P E 2 b を識別する情報である P E 番号とこれらのプロセッサエレメント 2 の実行許可状態を識別する情報である実行許可フラグとを対応付けて構成され、P E 実行許可部 3 4 が、この割込管理テーブル 3 3 a の実行許可フラグを設定するようになっている。

【 0 0 2 8 】

なお、図 3 に示す動作許可テーブル 3 2 中において、P E 番号「1」が P E 2 a を示すものであり、P E 番号「2」が P E 2 b を示すものである。

また、動作許可テーブル 3 2 においては、実行許可フラグとして、「0」もしくは「1」が設定されるようになっており、特に、2 以上のプロセッサエレメント 2 のうち、いずれか 1 つのプロセッサエレメント 2 にだけ「1」が設定され、これ以外のプロセッサエレメント 2 には実行許可フラグとして「0」が設定されるようになっている。すなわち、割込管理部 3 1 は、常に P E 2 a もしくは P E 2 b のいずれか一方だけに実行許可フラグ「1」を設定し、他方には「0」を設定するようになっている。実行許可フラグ「1」を設定されているプロセッサエ

レメント（実行許可プロセッサエレメント）2が、処理の実行を許可されたものであり、実行許可フラグ「0」が設定されているプロセッサエレメント2は、動作許可の出ていない状態であり停止状態（プロセッサの低消費電力モードで待機など）となる。

#### 【0029】

なお、この動作許可テーブル32により実行許可フラグ「1」が設定されたプロセッサエレメント2が、被選択プロセッサエレメント2に相当するものである。

割込管理テーブル33aは、図4に示すように、“割込番号”，“割込対象”，“ベクタ番号”および“優先順位”をそれぞれ対応付けて管理するものである。“割込番号”は、PE制御部3に入力される割込信号を特定するための番号であり、“割込対象”は、その“割込番号”によって要求される処理を行なうべきプロセッサエレメント2（PE2a，PE2b）を特定するための情報である。なお、本第1実施形態においては、この“割込対象”にはすべて「active」が設定されている。これは、上述した動作許可テーブル32において実行許可フラグ「1」が設定されているプロセッサエレメント2に対してプロセッサエレメント用割込信号を入力することを意味する。

#### 【0030】

“ベクタ番号”はPE2aやPE2bに入力されるプロセッサエレメント用割込信号の割込番号を示すものであり、“優先順位”は“割込番号”によって特定される各割込処理の優先順位を示すものである。そして、割込管理部31が、PE制御部3に入力された信号に基づいて、この割込管理テーブル33aを形成するようになっている。

#### 【0031】

PE実行許可部34は、上述の如く動作許可テーブル32に実行許可フラグを設定することにより各プロセッサエレメント2の動作・停止を制御するものである。又、このPE実行許可部34には、PE2aおよびPE2bから切替要求信号が入力されるようになっていて、PE実行許可部34は、この切替要求信号に基づいて、動作許可テーブル32の実行許可フラグを設定することにより、各プ

ロセッサエレメント 2 の動作・停止を制御するようになっている。

【 0 0 3 2 】

具体的には、P E 実行許可部 3 4 は、P E 2 a もしくは P E 2 b からの切替要求に応じて、切替要求を行なったプロセッサエレメント 2 について、動作許可テーブル 3 2 の実行許可フラグに「0」を設定することにより禁止状態にするとともに、切替先として要求された側のプロセッサエレメント 2 について、動作許可テーブル 3 2 の実行許可フラグに「1」を設定することにより許可状態にする。これにより、切替要求を行なったプロセッサエレメント 2 は待機（停止）状態となり、切替要求が行なわれたプロセッサエレメント 2 は停止状態から動作状態へと遷移する。

【 0 0 3 3 】

割込管理部 3 1 は、上述の如く外部からの割込信号に基づいて割込管理テーブル 3 3 a を形成するものであり、これにより、各プロセッサエレメント 2（P E 2 a，P E 2 b）に対する割込を制御するようになっており、動作許可テーブル 3 2 において動作許可フラグ「1」が設定された被選択プロセッサエレメント 2 に対して、プロセッサエレメント用割込信号を出力するようになっている。

【 0 0 3 4 】

すなわち、P E 制御部 3 は、動作許可テーブル 3 2 および割込管理テーブル 3 3 a に基づいて、P E 2 a および P E 2 b の切替を制御するようになっており、この際、一つのプロセッサエレメント 2（アクティブ P E）だけが動作し、それ以外のプロセッサエレメント 2 はその処理を停止するようになっている。

なお、本第 1 実施形態においては、プロセッサエレメント 2 の切替えは、P E 2 a もしくは P E 2 b からの切替要求信号によって行なわれる。この切替要求は、各プロセッサエレメント 2 に専用命令を追加することによって行なっても良く、又、割込復帰命令を発行することによって行なっても良く、いずれの場合でも、これらの命令が発行された時に、P E 制御部 3 の P E 実行許可部 3 4 へ切替え要求信号が入力される。

【 0 0 3 5 】

そして、本実施形態においては、情報処理装置の C P U（Central Processing



Unit) が、コンピュータ読取可能な記録媒体 (例えば、メモリ、磁気記憶装置、フロッピーディスク、メモリカード、光磁気記憶装置、CD-ROM、CD-R、CD-RW、DVD、DVD-R、DVD-RW等) に格納されたプログラム (マルチプロセッサ制御プログラム) を実行することにより、PE制御部3 (割込管理部31、動作許可テーブル32、割込管理テーブル33aおよびPE実行許可部34) として機能するようになっている。

## 【0036】

なお、かかるPE制御部3としての機能の実現手法は、上記の如くCPUがプログラムを実行することに限定されるものではなく、PE制御部3としての機能を有するハードウェアをそなえることによって実現してもよい。

共有メモリ部4は、PE制御部3による被選択プロセッサエレメント2の切替時に、上述した共有情報を格納するものである。この共有メモリ部4には、PE2aおよびPE2bによって共有情報を書き込まれるようになっており、又、この書き込まれた共有情報をPE2aおよびPE2bによって読み出されるようになっている。すなわち、PE2aとPE2bとは、この共有メモリ部4を介して共有情報を受け渡すようになっている。なお、この共有メモリ部4は、例えば、RAM (Random Access Memory) 等によって構成されている。

## 【0037】

上述の如く構成された本発明の第1実施形態のマルチプロセッサシステム1aにおける、割り込み処理を除く通常の処理手順を、図6を参照しながら、図5に示すシーケンス図に従って説明する。

なお、図6は本第1実施形態のマルチプロセッサシステム1aにおけるライブラリ動作を説明するための図である。

## 【0038】

OS上で稼働するメインプログラムP1 (図6参照) がPE2aによって実行され、このメインプログラムP1から、PE2bによる関数D1 (DSP処理; 図6参照) の処理が要求 (関数コール) されると (図5のt1参照)、図6に示すライブラリL1が実行され、PE2aから共有情報が共有メモリ部4に格納 (リソース退避) されるとともに、PE2aからPE2bへの切替えが行なわれる

(図5のt2～t3参照)

次に、ライブラリL1において、PE2bが共有メモリ部4から共有情報を読み出した後(リソース復元;図5のt4～t5参照)、ライブラリL1による関数呼び出しが行なわれてPE2bにより関数D1が実行される(図5のt6～t7参照)。

【0039】

PE2bによる関数D1の処理が終了すると、ライブラリL1において、PE2bから戻り値等の共有情報が共有メモリ部4に格納されるとともに(リソース退避)、PE2bからPE2aへの切替えが行なわれ(図5のt8～t9参照)、その後、PE2aが共有メモリ部4から共有情報を読み出してリソース復元を行なう(図5のt10～t11参照)。これにより、ライブラリL1による処理が終了され、PE2aによるメインプログラムP1の処理が引き続き行なわれる(図5のt12参照)。

【0040】

このように、本第1実施形態においては、ライブラリL1を介してメインプログラムP1から関数コールによってPE2bによる関数処理(DSP処理)に移行し、このDSP処理の終了によって関数処理からメインプログラムP1に復帰してMPU処理が再開され、見かけ上は1本のプログラムにおいてPE2aによる処理からPE2bによる処理への切替えが行なわれる。また、この際、PE2bによる処理が行なわれている間はPE2aは停止しており、同様に、PE2aによる処理が行なわれている間はPE2bは停止している。

【0041】

そして、プログラミング時において、例えば#pragma等のコンパイラ指示によって、コンパイラにDSP処理を行なう関数をDSP処理関数と判断させることにより、このDSP処理を行なう関数がDSP命令としてコンパイルされる。

また、コンパイラは、このDSP関数をコールしたり関数から復帰したりする為に、プログラム下位層でプロセッサ間コーリングライブラリ(ライブラリL1)を経由するようにコード生成を行なう。図6に示すように、ライブラリL1はプロセッサ切替え命令(例えば、switch DSP, switch MPU)を含み、共有メモリ

部 4 を利用して関数への引数渡しや戻り値の引渡しを行なわせる。

【 0 0 4 2 】

このように、本発明の第 1 実施形態としてのマルチプロセッサシステム 1 a を実現するためのプログラム（マルチプロセッサ制御プログラム）を作成するに際して、プログラマは、DSP 命令を使って DSP に処理させたい信号処理ルーチン等を、一続きのプログラムの中で部分的にコンパイラへ指示するだけで、MPU と DSP とを効率良く利用することができる。

【 0 0 4 3 】

次に、本第 1 実施形態のマルチプロセッサシステムにおける割り込みが発生した場合の制御手法を、図 7 に示すフローチャート（ステップ A 1 0 ～ A 1 1 0 ）に従って説明する。

例えば、PE 2 a および PE 2 b のいずれか一方のプロセッサエレメント 2 による処理中に割り込みが発生すると、割込管理部 3 1 は、割込管理テーブル 3 3 a を参照して、その割込処理に対応するプロセッサエレメント 2 を特定するとともに（ステップ A 1 0 ）、動作許可テーブル 3 2 を参照して、“Active”となっているプロセッサエレメント（被選択プロセッサエレメント）2 に対してプロセッサエレメント用割込信号を入力して（ステップ A 2 0 ）、そのプロセッサエレメント 2 （以下、アクティブ PE という場合もある）の割込処理ルーチンへ移行する（ステップ A 3 0 ）。なお、これらのステップ A 1 0 および A 2 0 の処理が割込管理部 3 1 によって行なわれるようになっている。

【 0 0 4 4 】

プロセッサエレメント用割込信号を受けたアクティブ PE は、その割込ハンドラによって、その割込要求の処理を行なうべきプロセッサエレメント 2 を判断して、自らがその処理を行なうべきか否かを判断する（ステップ A 4 0 ）。ここで、自らがその処理を行なうべきであると判断した場合には（ステップ A 4 0 の YES ルート参照）、アクティブ PE は通常割込処理を行なう（ステップ A 5 0 ）。

【 0 0 4 5 】

この通常割込処理は、その割込処理の優先順位に応じて、実行中の処理に関す

る必要な情報（プログラムカウンタやスタックポインタ、関数の引数、演算結果等）を、図示しない記憶部に一時的に格納した後に、要求された割込処理を実行することにより行なわれる。

なお、この通常割込処理に使用される記憶部は、RAM等により構成されている。なお、共有メモリ部4の一部をこの記憶部として用いてもよい。

#### 【0046】

そして、アクティブPEは、この通常割込処理を行なった後に、ステップA50において記憶部に一時的に格納した情報を再度取得して、割込処理を行なう前の状態に復帰し（ステップA60）、処理を終了する。

一方、アクティブPEは、その割込ハンドラによって、要求された処理は自らが行なうべきものではないと判断した場合には（ステップA40のNOルート参照）、共有情報を共有メモリ部4に格納してリソース退避を行なう（ステップA70）。又、この際、アクティブPEは、PE実行許可部34に対して切替要求信号を出力する。

#### 【0047】

PE実行許可部34は、アクティブPEから入力された切替要求信号に基づいて、動作許可テーブル32の実行許可フラグの設定を行ない、アクティブPEの動作を停止させるとともに、要求された処理を行なうべきプロセッサエレメント2（以下、処理対象PEという場合がある）の動作を許可する。

処理対象PEは、タスクを切替えるための設定を行ない、共有メモリ部4から共有情報を取得（リソース復帰処理）して（ステップA80）、割り込み処理ルーチンから復帰する際にタスクをスイッチングする。このタスクのスイッチング処理の一環として処理対象PEへの切替えが行なわれる。

#### 【0048】

その後、処理対象PEは通常割込処理を行なう（ステップA90）。なお、この通常割込処理は、ステップA50においてアクティブPEが行なう処理と同様のものである。処理対象PEが、タスク処理の終了時にアクティブPEへ切替えを行なうことにより、アクティブPE側では次のタスクへの切替えを行ない処理を継続する。なお、この処理手順は、OSのタスク管理方式などの実装方法に依

存する。

【0049】

また、この際、処理対象PEは、PE実行許可部34に対して切替要求信号を出力する。PE実行許可部34は、処理対象PEから入力された切替要求信号に基づいて動作許可テーブル32の実行許可フラグの設定を行ない、処理対象PEの動作を停止させるとともにアクティブPEの動作を許可する。又、処理対象PEは、共有情報を共有メモリ部4に退避させる（ステップA100）。

【0050】

アクティブPEは、共有メモリ部4から共有情報を取得した後（ステップA110）、ステップA60に移行して処理を終了する。

このように、本発明の第1実施形態としてのマルチプロセッサシステム1aによれば、例えば、MPUやDSP等の互いに異なる機能を有する2以上のプロセッサエレメント2（本実施形態ではPE2aおよびPE2bの2つ）の内から一つのプロセッサエレメント2を被選択プロセッサエレメント2として選択して動作させることができ、この際に、1つの連続したプログラム中においてPE2aの処理とPE2bの処理とを混在させて効率よく処理することができる。

【0051】

また、PE動作許可部34により、切替え後のプロセッサエレメント2を停止させるようになっているので、一つのプロセッサエレメント2（アクティブPE）だけが動作し、それ以外のプロセッサエレメント2はその処理を停止するようになっているので、消費電力を低減させることができるほか、プログラムが一続きになるように記述できるので、プログラミングが容易になり、又、メンテナンス性が向上する。

【0052】

さらに、アクティブPEから処理対象PEへの切替えを、プログラム中から命令（切替要求命令）を発行することによって行なうことができ、1つの連続したプログラムとして記述することができる。

また、被選択プロセッサエレメント2の切替え時に、共有情報を切替え前の被選択プロセッサエレメント2から共有メモリ部4に格納し、この共有情報の格納

後に、切替え前の被選択プロセッサエレメント2を停止させるとともに、共有メモリ部4に格納された共有情報を用いて、切替え後の被選択プロセッサエレメント2を動作させることにより、共有情報を切替え後の被選択プロセッサエレメント2に対して容易且つ確実に受け渡すことができ被選択プロセッサエレメント2の切替えを容易且つ確実に行なうことができる。

【0053】

さらに、プロセッサエレメント2が、割り込みによってプロセッサエレメント2に対して要求された処理が、このプロセッサエレメント2以外のプロセッサエレメント2が行なうべきものである場合に、PE制御部3（PE実行許可部34）に対してプロセッサエレメント2の切替要求信号を出力することにより、プロセッサエレメント2の切替えを容易且つ確実に行なうことができる。又、この際、切替え前のプロセッサエレメント2において、切替えの発生を把握することもできる。

【0054】

また、PE制御部3が、動作許可テーブル32および割込管理テーブル33aをそなえとともに、これらの動作許可テーブル32および割込管理テーブル33aに基づいて、被選択プロセッサエレメント2の切替えを制御することによって、一つのプログラム中においてPE2aの処理とPE2bの処理とを混在させる事ができ、更に、被選択プロセッサエレメント2の切替えを容易に制御することができる。

【0055】

さらに、PE制御部3（割込管理部31）が、動作許可テーブル32において実行許可フラグ「1」が設定された被選択プロセッサエレメント2に対して、プロセッサエレメント用割込信号を出力することにより、各被選択プロセッサエレメント2の動作・停止を容易且つ確実に制御することができる。

また、PE制御部3（割込管理部31）が外部から信号を受信すると、この信号を処理すべき一のプロセッサエレメント2を2以上（本実施形態では2つ）のプロセッサエレメントの中から選択して動作させ、更に、この一のプロセッサエレメント2以外のプロセッサエレメント2の動作を停止させることにより、消費

電力を低減することができるほか、見かけ上は1本のプログラム中でPE 2 aによる処理からPE 2 bによる処理への切替えを行なうことができ、プログラムのメンテナンス性が向上する。

【0056】

さらに、各プロセッサエレメント2が、互いに異なる機能を有することにより、各プロセッサエレメント2に好適な処理を行なわせることができ、処理効率を向上させることができる。

特に、2以上のプロセッサエレメント2として、MPUとDSPとをそなえて構成することにより、MPUによってOSによる処理を行なわせるとともに、音声や画像、動画の処理や入出力処理等のリアルタイム性が求められる処理をDSPによって行なわせることができる。

【0057】

#### (B) 第2実施形態の説明

図8は本発明の第2実施形態としてのマルチプロセッサシステムの割込管理テーブル33bの例を示す図である。本第2実施形態のマルチプロセッサシステムは、第1実施形態のマルチプロセッサシステム1aにおいて、図4に示す割込管理テーブル33aに代えて図8に示す割込管理テーブル33bをそなえるものであり、この割込管理テーブル33b以外については、上述した第1実施形態のマルチプロセッサシステム1aとほぼ同一の構成をそなえている。

【0058】

本第2実施形態としてのマルチプロセッサシステムは、既存のシングルプロセッサ用OSをそなえており、PE 2 aによって、各種制御をこのOSの管理下で行なうとともに、必要に応じてPE 2 bによるDSP処理を実行するようになっている。

すなわち、本第2実施形態のマルチプロセッサシステムにおいては、PE制御部3が、2以上のプロセッサエレメント2の内の一のプロセッサエレメント2 (PE 2 a) を優先的に動作させ、必要に応じて、このPE 2 aに代えてPE 2 bを動作させるようになっている。

【0059】

本第2実施形態のマルチプロセッサシステムにおいても、第1実施形態と同様に、PE制御部3は、動作許可テーブル32（図3参照）および割込管理テーブル33bを参照して、割り込みが発生した場合の制御を行なうようになっている。

そして、本第2実施形態においては、図8に示すように、OS処理を行なうPE2aに全ての割り込み処理が入力され、このPE2aの割込ハンドラによって全ての割込処理が制御されるようになっている。これにより、OS側において全ての割込処理を管理することができる。

#### 【0060】

上述の如く構成された本発明の第2実施形態のマルチプロセッサシステムにおける割り込み以外の通常処理は、PE2aによるプログラム処理によって行なわれる。そして、このプログラム処理においてDSP処理が必要となった場合に、第1実施形態のマルチプロセッサシステム1aと同様に、図5に示すシーケンス図に従った処理が行なわれる。すなわち、メインプログラムP1における関数コールにより、ライブラリL1に従ってDSP処理（DSP命令で記述された関数）へ切替えられ、このDSP処理の終了後、ライブラリL1に従ってメインプログラムP1に復帰する。

#### 【0061】

すなわち、本第2実施形態のマルチプロセッサシステムにおいても、見かけ上は1本のプログラム中でPE2aによる処理からPE2bによる処理への切替えが行なわれ、又、この際、PE2bによる処理が行なわれている間はPE2aは停止しており、同様に、PE2aによる処理が行なわれている間はPE2bは停止している。

#### 【0062】

次に、本第2実施形態のマルチプロセッサシステムにおける割り込みが発生した場合の制御手法を、図9に示すフローチャート（ステップB10～B150）に従って説明する。

例えば、PE2aおよびPE2bのいずれか一方のプロセッサエレメント2による処理中に割り込みが発生すると、割込管理部31は、割込管理テーブル33



を参照して、その割込信号の割込番号に対応するプロセッサエレメント 2 を特定する（ステップ B 1 0）。そして、割込管理部 3 1 は、そのプロセッサエレメント 2 に対してプロセッサエレメント用割込信号を入力する（ステップ B 2 0）。なお、本第 2 実施形態のマルチプロセッサシステムにおいては、図 8 に示すように、P E 2 a に全ての割込信号が入力されるようになっている。

## 【 0 0 6 3 】

次に、割込管理部 3 1 は、動作許可テーブル 3 2 を参照して、割込処理を行なうプロセッサエレメント 2 が動作中（アクティブ）であるか否かを判断し（ステップ B 3 0）、このプロセッサエレメント 2 が動作中である場合には（ステップ B 3 0 の Y E S ルート参照）、そのプロセッサエレメント 2（本第 2 実施形態では P E 2 a）の割込処理ルーチンに移行する（ステップ B 5 0）。

## 【 0 0 6 4 】

また、割込を行なうプロセッサエレメント 2（P E 2 a）が停止中である場合には（ステップ B 3 0 の N O ルート参照）、P E 実行許可部 3 4 は、動作中のプロセッサエレメント 2（P E 2 b）を停止させ、P E 2 a に対して動作許可を行なうとともに（ステップ B 4 0）、P E 2 a に対してプロセッサエレメント用割込信号を出力して P E 2 a を起動してステップ B 5 0 に移行する。

## 【 0 0 6 5 】

なお、これらのステップ B 1 0 ～ B 4 0 の処理が、割込管理部 3 1 によって行なわれるようになっている。

プロセッサエレメント用割込信号を受けた P E 2 a は、その割込ハンドラによって、その割込要求の処理を行なうべきプロセッサエレメント 2 を判断して、自らがその処理を行なうべきか否かを判断する（ステップ B 6 0）。ここで、自らがその処理を行なうべきであると判断した場合には（ステップ B 4 0 の Y E S ルート参照）、P E 2 a は通常割込処理を行なう（ステップ B 7 0）。

## 【 0 0 6 6 】

この通常割込処理は、その割込処理の優先順位に応じて、実行中の処理に関する必要な情報（プログラムカウンタやスタックポインタ、関数の引数、演算結果等）を、記憶部（図示省略）に一時的に格納した後に、要求された割込処理を実

行することにより行なわれる。又、この割込処理として、タスクの切替処理を行なうこともある。

## 【0067】

そして、PE 2 a は、この通常割込処理を行なった後に、ステップ B 20 において割込入力が行なわれた際に、自らが動作中であったか否かを判断して（ステップ B 80）、動作中であって場合には（ステップ B 80 の YES ルート参照）、ステップ B 70 において記憶部に一時的に格納した情報を再度取得して、割込処理を行なう前の状態に復帰し（ステップ B 90）、処理を終了する。

## 【0068】

また、PE 2 a は、ステップ B 20 において割込入力が行なわれた際に、自らが停止中であった場合には（ステップ B 80 の NO ルート参照）、PE 実行許可部 34 に対して PE 2 b への切替え要求を行ない（ステップ B 150）、処理を終了する。

一方、PE 2 a は、その割込ハンドラによって、要求された処理は自らが行なうべきものではないと判断した場合には（ステップ B 60 の NO ルート参照）、共有情報を共有メモリ部 4 に格納してリソース退避を行なう（ステップ B 100）。又、この際、PE 2 a は、PE 実行許可部 34 に対して切替要求信号を出力する。

## 【0069】

PE 実行許可部 34 は、PE 2 a から入力された切替要求信号に基づいて、動作許可テーブル 32 の実行許可フラグの設定を行ない、PE 2 a の動作を停止させるとともに、要求された処理を行なうべきプロセッサエレメント 2（本第2実施形態では PE 2 b）の動作を許可する。

PE 2 b は、共有メモリ部 4 から共有情報を取得（リソース復帰処理）した後（ステップ B 110）、通常割込処理を行なう（ステップ B 120）。なお、この通常割込処理は、ステップ B 70 において PE 2 a が行なう処理と同様のものである。また、この処理手順は、OS のタスク管理方式などの実装方法に依存する。

## 【0070】

割込処理が終了すると、P E 2 b は、P E 実行許可部 3 4 に対して、切替要求信号を出力し、P E 実行許可部 3 4 は、この P E 2 b から入力された切替要求信号に基づいて、動作許可テーブル 3 2 の実行許可フラグの設定を行ない、P E 2 b の動作を停止させるとともに、P E 2 a の動作を許可する。又、P E 2 b は、共有情報を共有メモリ部 4 に退避させる（ステップ B 1 3 0）。

## 【 0 0 7 1 】

アクティブ P E 2 a は、共有メモリ部 4 から共有情報を取得した後（ステップ B 1 4 0）、ステップ B 8 0 に移行する。

このように、本発明の第 2 実施形態としてのマルチプロセッサシステムによれば、上述した第 1 実施形態と同様の作用効果を得ることができる他、P E 制御部 3 が、2 以上のプロセッサエレメント 2 の内の一のプロセッサエレメント 2 を優先的に動作させ、必要に応じて、一のプロセッサエレメント 2 に代えて他のプロセッサエレメント 2 を動作させることにより、優先的に動作させるプロセッサエレメント 2 において、他のプロセッサエレメント 2 への切替えを把握することができる。すなわち、全ての割込処理を P E 2 a の割込ハンドラで管理することができ、デバッグ等のメンテナンスやプログラミングが容易になり利便性が向上する。

## 【 0 0 7 2 】

## （C）第 3 実施形態の説明

図 1 0 は本発明の第 3 実施形態としてのマルチプロセッサシステムの割込管理テーブルの例を示す図である。本第 3 実施形態のマルチプロセッサシステムは、第 1 実施形態のマルチプロセッサシステム 1 a において、図 4 に示す割込管理テーブル 3 3 a に代えて図 1 0 に示す割込管理テーブル 3 3 c をそなえるものであり、この割込管理テーブル 3 3 c 以外の構成は、上述した第 1 実施形態のマルチプロセッサシステム 1 a とほぼ同一の構成をそなえている。

## 【 0 0 7 3 】

さて、一般に、情報処理端末においては、O S が各タスクを管理し、割り込みに応じて各タスクのスイッチング等を行なうのであるが、このようなタスクのスイッチングには大きなオーバーヘッドを伴う。

一方、DSP処理においては、例えば、割り込みに応じてフラグやステータスを書き換えるだけの処理のように、プログラムとしては数ステップで実行可能な処理が存在する。

## 【 0 0 7 4 】

これらの簡単で負荷が小さい処理（以下、微少処理という場合もある）の全てをMPU側に割り込みとして入力して、毎回MPUにその処理内容を判断させ、更に、DSPに切替えて処理を行なう事はシステムのパフォーマンスの低下を招く。

本第3実施形態のマルチプロセッサシステムでは、OSが介在する必要性のない処理や、OSが介在するとパフォーマンスが大きく低下するような処理（特に、処理単位が小さく、割り込み頻度の高い処理）については、MPUを介さずに直接DSPに割り込みとして入力し、このDSPの割込ハンドラによって処理を行なう。

## 【 0 0 7 5 】

このDSPのハンドラとしては、必要なDSP処理を行なって復帰するだけの機能を最低限そなえていれば良い。又、微少処理をDSPに実行させている間には、更なる割り込みがDSPに入力されないように、多重割り込みを禁止することが望ましい。これにより、中断されていた元の処理に確実かつ速やかに復帰できる。

## 【 0 0 7 6 】

本第3実施形態としてのマルチプロセッサシステムは、既存のシングルプロセッサ用OSをそなえており、PE2a側の割り込みハンドラを主として動作させて各種制御をPE2aによってこのOSの管理下で行なうとともに、OSによるタスク管理等の手順を踏む必要のない微小処理については、PE2bに割り込みとして処理させるようになっている。

## 【 0 0 7 7 】

すなわち、本第3実施形態のマルチプロセッサシステムにおいては、図10に示すように、PE2aとPE2bとの両方に割込管理部31からのプロセッサエレメント用割込信号が入力されるようになっているのである。

また、割込管理テーブル 3 3 c は、図 1 0 に示すように、各割り込み要因（割込番号によって特定する）について、対応すべきプロセッサエレメント 2 を予め登録して構成されている。

【 0 0 7 8 】

そして、P E 制御部 3（割込管理部 3 1）は、この割込管理テーブル 3 3 c および動作許可テーブル 3 2 を参照して、各プロセッサエレメント 2 に対する割込処理を制御するようになっている。

上述の如く構成された本発明の第 3 実施形態のマルチプロセッサシステムにおける割り込み以外の通常処理においては、図 5 および図 6 を用いて上述した第 2 実施形態のマルチプロセッサシステムと同様であるので、その説明は省略する。

【 0 0 7 9 】

また、本第 3 実施形態のマルチプロセッサシステムにおいて、P E 2 a（M P U）が処理すべき割込処理が入った場合の制御手法は、図 9 を用いて上述した第 2 実施形態のマルチプロセッサシステムにおける制御手法と同様であるので、その説明は省略する。

本第 3 実施形態のマルチプロセッサシステムにおける P E 2 b（D S P）が処理すべき割込処理（微少処理）が入った場合の制御手法を、図 1 1 に示すフローチャート（ステップ C 1 0 ～ C 1 3 0）に従って説明する。

【 0 0 8 0 】

例えば、P E 2 a および P E 2 b のいずれか一方のプロセッサエレメント 2 による処理中に割り込みが発生すると、割込管理部 3 1 は、割込管理テーブル 3 3 を参照して、その割込信号の割込番号に対応するプロセッサエレメント 2 を特定する（ステップ C 1 0）。そして、割込管理部 3 1 は、そのプロセッサエレメント 2（本説明では P E 2 b（D S P））に対してプロセッサエレメント用割込信号を入力する（ステップ C 2 0）。

【 0 0 8 1 】

次に、割込管理部 3 1 は、動作許可テーブル 3 2 を参照して、割込処理を行なう P E 2 b が動作中（アクティブ）であるか否かを判断し（ステップ C 3 0）、P E 2 b が動作中である場合には（ステップ C 3 0 の Y E S ルート参照）、その

PE 2 b の割込処理ルーチンに移行する（ステップ C 5 0）。

また、PE 2 b が停止中である場合には（ステップ C 3 0 の NO ルート参照）、PE 実行許可部 3 4 は、動作中の PE 2 a を停止させるとともに、PE 2 b に動作許可を行なうとともに（ステップ C 4 0）、PE 2 b に対してプロセッサエレメント用割込信号を出力して PE 2 b を起動してステップ B 5 0 に移行する。

【0082】

なお、これらのステップ C 1 0 ～ C 4 0 の処理が、割込管理部 3 1 によって行なわれるようになっている。

PE 2 b においては、多重割込を禁止した後（ステップ C 6 0）、その割込処理の優先順位に応じて、実行中の処理に関する必要な情報（プログラムカウンタやスタックポインタ、関数の引数、演算結果等；これらの情報を総称してリソースともいう）を、記憶部（図示省略）に一時的に格納してリソースの退避を行なう（ステップ C 7 0）。

【0083】

そして、PE 2 b は、その微少処理を行なった後（ステップ C 8 0）、記憶部に退避したリソースを読み出す（ステップ C 9 0）。更に、PE 2 b は、この通常割込処理を行なった後に、ステップ C 2 0 において割込入力が行なわれた際に、自らが動作中であったか否かを判断して（ステップ C 1 1 0）、動作中であった場合には（ステップ C 1 1 0 の YES ルート参照）、ステップ C 7 0 において記憶部に一時的に格納した情報を再度取得して、割込処理を行なう前の状態に復帰し（ステップ C 1 3 0）、処理を終了する。

【0084】

また、PE 2 b は、ステップ C 2 0 において割込入力が行なわれた際に、自らが停止中であった場合には（ステップ C 1 1 0 の NO ルート参照）、PE 実行許可部 3 4 に対して PE 2 a への切替要求を行ない（ステップ C 1 2 0）、処理を終了する。

このように、本発明の第 3 実施形態としてのマルチプロセッサシステムによれば、上述した第 2 実施形態と同様の作用効果を得ることができる他、微少処理については、直接 DSP（PE 2 b）に割り込みとして入力して、この DSP の割

込ハンドラによって処理を行なうようになっているので、MPU（PE 2 a）の割込ハンドラにより処理内容の判断および切替処理を行なう必要がなく、これらの処理によるオーバーヘッドを削減することができ、処理速度を向上させることができる。

【0085】

また、微少処理をPE 2 bが実行している間には、更なる割り込みがDSPに入力されないように、多重割り込みを禁止することにより、中断されていた元の処理に確実に速やかに復帰できる。

#### （D）第4実施形態の説明

図12および図13は本発明の第4実施形態としてのマルチプロセッサシステム1bの構成を説明するためのもので、図12はその機能構成を示すブロック図、図13は第4実施形態としてのマルチプロセッサシステム1bにおける機能構成および信号の流れを説明するためのブロック図である。

【0086】

本第4実施形態のマルチプロセッサシステム1bにおいては、図12および図13に示すように、第1実施形態のマルチプロセッサシステム1aにおいて各プロセッサエレメントからPE実行許可部34に入力されていた切替要求信号に代えて、各プロセッサエレメント2からPE制御部3の割込管理部31に、切替制御用割込信号として切替要求信号が入力されるようになっている。

【0087】

なお、図中、既述の符号と同一の符号は同一もしくは略同一の部分を示しているので、その説明は省略する。

一般に、ソフトウェアによる割り込み（以下、ソフトウェア割り込み）はプロセッサエレメントの内部で閉じた処理として行なわれている。例えば、ソフトウェア割り込み命令をきっかけとして割り込み処理へとプログラム分岐し、この割り込み処理終了後に、割り込み復帰命令によって、中断されていた処理に復帰する。

【0088】

本第4実施形態のマルチプロセッサシステム1bにおいては、各プロセッサエ

レメント 2 が P E 制御部 3 に対してプロセッサエレメント 2 の切替え要求を行なう際に、P E 制御部 3 へ切替制御用割込信号を出力するようになっている。

また、P E 制御部 3 においては、各プロセッサエレメント 2 からの切替制御割込信号が入力されると、その切替制御割込信号に基づいて割込管理部 3 1 が各プロセッサエレメント 2 の制御を行なうようになっている。具体的には、P E 制御部 3 は、P E 実行許可部 3 4 によって、その切替制御割込信号を出力したプロセッサエレメント 2 を動作禁止状態にするとともに、切替え後のプロセッサエレメント 2 を動作許可状態にする。

【 0 0 8 9 】

このように、本発明の第 4 実施形態としてのマルチプロセッサシステム 1 b においては、上述した第 1 実施形態～第 3 実施形態において、各プロセッサエレメント 2 から P E 制御部 3 の P E 実行許可部 3 4 に対して切替要求信号を送信して行なっていたプロセッサエレメント 2 の切替え要求を、P E 制御部 3 に入力された割り込みの一つとして処理することができる。

【 0 0 9 0 】

これにより、P E 制御部 3 においては、プロセッサエレメント 2 の切替え処理を他の処理と一元的に管理することができる。又、プロセッサエレメント 2 の内部的には、通常、内部で閉じているソフト割り込み命令を、あたかも他プロセッサへの内部割込命令として動作することができる。

(E) 第 5 実施形態の説明

本発明の第 5 実施形態としてのマルチプロセッサシステムは、上述した第 1 実施形態～第 4 実施形態の各マルチプロセッサシステムに、P E 制御部 3 の切替え機能を無効化して、各プロセッサエレメント 2 を同時に動作させうる無効化部（図示省略）をそなえて構成されている。

【 0 0 9 1 】

そして、情報処理装置の C P U (Central Processing Unit) が、コンピュータ読取可能な記録媒体（例えば、メモリ、磁気記憶装置、フロッピーディスク、メモリカード、光磁気記憶装置、C D - R O M , C D - R , C D - R W , D V D , D V D - R , D V D - R W 等）に格納されたプログラム（マルチプロセッサ制



御プログラム) を実行することにより、無効化部として機能するようになっている。

なお、この無効化部としての機能の実現手法は、上記の如くCPUがプログラムを実行することに限定されるものではなく、無効化部としての機能を有するハードウェアをそなえることによって実現してもよい。

【0092】

この無効化部は、PE制御部3のPE実行許可部34の機能を無効化することによりその機能を実現するようになっており、例えば、ユーザがPE制御部3の切替え機能の無効化を任意に設定することができるようになっている。

そして、PE実行許可部34の機能が無効化されると、各プロセッサエレメント2は、PE実行許可部34からの実行停止の制御を受けずに、それぞれ独立して同時実行的に動作可能となる(無効モード)。

【0093】

このように、本発明の第5実施形態のマルチプロセッサシステムにおいては、無効化部によって無効化モードに設定することにより、全てのプロセッサエレメント2を同時に動作させることができ、例えば、高い処理性能が要求される処理を行なう場合に、高速に処理を行なうことができ、更に、本発明のマルチプロセッサシステムを柔軟に使用することができる。

【0094】

(F) その他

そして、本発明は上述した実施形態に限定されるものではなく、本発明の趣旨を逸脱しない範囲で種々変形して実施することができる。

例えば、上述した各実施形態では、プロセッサエレメント2を2つそなえた場合について説明しているが、これに限定されるものではなく、プロセッサエレメント2を3以上そなえてもよい。

【0095】

また、上述した各実施形態では、2以上のプロセッサエレメントとして、MPUとDSPとを用いた場合について説明しているが、これに限定さえるものではなく、これらのMPUやDSP以外のプロセッサエレメントを使用してもよい。

さらに、上述した各実施形態では、2以上のプロセッサエレメントとして、違う種類のプロセッサエレメント（MPUとDSP）を用いた場合について説明しているが、これに限定されるものではなく、例えば、2以上のMPUのように、同じ種類のプロセッサエレメントをそなえてもよい。

【0096】

また、上述した各実施形態では、共有メモリ部4や記憶部は、RAM（Random Access Memory）等によって構成されているが、これに限定されるものではなく、例えば、磁気記憶装置やメモリカード等の他の記憶媒体を用いてもよい。

なお、本発明の各実施形態が開示されていれば、当業者によって製造することが可能である。

【0097】

（G）付記

（付記1） 2以上のプロセッサエレメントと、

前記2以上のプロセッサエレメントの内から、一つのプロセッサエレメントを被選択プロセッサエレメントとして選択して動作させる制御部と、

該制御部による前記被選択プロセッサエレメントの切替え時に、切替え前の被選択プロセッサエレメントから切替え後の被選択プロセッサエレメントへ受け渡すべき情報を格納する記憶部とをそなえたことを特徴とする、マルチプロセッサシステム。

【0098】

（付記2） 該制御部が、前記被選択プロセッサエレメントの切替え時に、前記情報を前記切替え前の被選択プロセッサエレメントから該記憶部に格納させ、前記情報の格納後に、前記切替え前の被選択プロセッサエレメントを停止させるとともに、該記憶部に格納された前記情報を用いて、前記切替え後の被選択プロセッサエレメントを動作させることを特徴とする、付記1記載のマルチプロセッサシステム。

【0099】

（付記3） 該プロセッサエレメントに対して要求された該動作が当該プロセッサエレメント以外のプロセッサエレメントによって実行されるべきものである

場合に、該プロセッサエレメントが、該制御部に対して該プロセッサエレメントの切替要求信号を出力することを特徴とする、付記 1 または付記 2 記載のマルチプロセッサシステム。

【 0 1 0 0 】

(付記 4) 該切替要求信号が、切替制御用割込信号であることを特徴とする、付記 3 記載のマルチプロセッサシステム。

(付記 5) 該制御部が、外部から信号を受信すると、前記被選択プロセッサエレメントに対してプロセッサエレメント用割込信号を出力することを特徴とする、付記 1 ～付記 4 のいずれか 1 項に記載のマルチプロセッサシステム。

【 0 1 0 1 】

(付記 6) 該制御部が、前記 2 以上のプロセッサエレメントの内、処理の実行を許可されている実行許可プロセッサエレメントを指示するテーブルをそなえるとともに、該テーブルに基づいて、前記被選択プロセッサエレメントの切替を制御することを特徴とする、付記 1 ～付記 5 のいずれか 1 項に記載のマルチプロセッサシステム。

【 0 1 0 2 】

(付記 7) 該制御部が、該テーブルを参照し、該テーブルによって指示される前記実行許可プロセッサエレメントを前記切替後の被選択プロセッサエレメントとして選択し、該実行許可プロセッサエレメントに対して、プロセッサエレメント用割込信号を出力することを特徴とする、付記 6 記載のマルチプロセッサシステム。

【 0 1 0 3 】

(付記 8) 該制御部が、前記 2 以上のプロセッサエレメントの内の一のプロセッサエレメントを優先的に動作させ、必要に応じて、該一のプロセッサエレメントに代えて他のプロセッサエレメントを動作させることを特徴とする、付記 1 ～付記 7 のいずれか 1 項に記載のマルチプロセッサシステム。

(付記 9) 前記 2 以上のプロセッサエレメントが、互いに異なる機能を有することを特徴とする、付記 1 ～付記 7 のいずれか 1 項に記載のマルチプロセッサシステム。

【 0 1 0 4 】

(付記 1 0) 該制御部が、外部から信号を受信すると、該信号を処理すべき一のプロセッサエレメントを前記 2 以上のプロセッサエレメントの内から選択して動作させることを特徴とする、付記 9 記載のマルチプロセッサシステム。

(付記 1 1) 前記 2 以上のプロセッサエレメントとして、MPU (Micro Processing Unit) と、DSP (Digital Signal Processor) とがそなえられたことを特徴とする、付記 9 または付記 1 0 記載のマルチプロセッサシステム。

【 0 1 0 5 】

(付記 1 2) 該制御部の切替機能が無効化して該 2 以上のプロセッサエレメントを同時に動作させうる無効化部をそなえることを特徴とする、付記 1 ～付記 1 1 のいずれか 1 項に記載のマルチプロセッサシステム。

(付記 1 3) 該記憶部に格納される前記情報が、プログラムカウンタ、関数の引数、関数の戻り値およびスタックポインタの内の少なくとも 1 つであることを特徴とする、付記 1 ～付記 1 2 のいずれか 1 項に記載のマルチプロセッサシステム。

【 0 1 0 6 】

(付記 1 4) 2 以上のプロセッサエレメントの内から一つのプロセッサエレメントを被選択プロセッサエレメントとして選択して動作させるように制御するマルチプロセッサ制御方法であって、

前記被選択プロセッサエレメントの切替え時に、切替え前の被選択プロセッサエレメントから切替え後の被選択プロセッサエレメントへ受け渡すべき情報を、前記切替え前の被選択プロセッサエレメントから記憶部に格納する格納ステップと、

前記情報の格納後に、該切替え前の被選択プロセッサエレメントを停止させるとともに、前記記憶部に格納された前記情報を用いて、切替え後の被選択プロセッサエレメントを動作させる切替ステップとをそなえたことを特徴とする、マルチプロセッサ制御方法。

【 0 1 0 7 】

(付記 1 5) 2 以上のプロセッサエレメントの内から一つのプロセッサエレ

メントを被選択プロセッサエレメントとして選択して動作させるように制御するマルチプロセッサ制御プログラムであって、

前記被選択プロセッサエレメントの切替え時に、切替え前の被選択プロセッサエレメントから切替え後の被選択プロセッサエレメントへ受け渡すべき情報を、前記切替え前の被選択プロセッサエレメントから記憶部に格納する格納ステップと、

前記情報の格納後に、該切替え前の被選択プロセッサエレメントを停止させるとともに、前記記憶部に格納された前記情報を用いて、切替え後の被選択プロセッサエレメントを動作させる切替ステップとを、コンピュータに実行させることを特徴とする、マルチプロセッサ制御プログラム。

【0108】

(付記16) 2以上のプロセッサエレメントの内から一つのプロセッサエレメントを被選択プロセッサエレメントとして選択して動作させるように制御するマルチプロセッサ制御プログラムを記録したコンピュータ読取可能な記録媒体であって、

該マルチプロセッサ制御プログラムが、

前記被選択プロセッサエレメントの切替え時に、切替え前の被選択プロセッサエレメントから切替え後の被選択プロセッサエレメントへ受け渡すべき情報を、前記切替え前の被選択プロセッサエレメントから記憶部に格納する格納ステップと、

前記情報の格納後に、該切替え前の被選択プロセッサエレメントを停止させるとともに、前記記憶部に格納された前記情報を用いて、切替え後の被選択プロセッサエレメントを動作させる切替ステップとを、該コンピュータに実行させることを特徴とする、マルチプロセッサ制御プログラムを記録したコンピュータ読取可能な記録媒体。

【0109】

【発明の効果】

以上詳述したように、本発明のマルチプロセッサシステム、マルチプロセッサ制御方法、マルチプロセッサ制御プログラムおよび同プログラムを記録したコン

コンピュータ読取可能な記録媒体によれば、以下の効果ないし利点がある。

(1) 被選択プロセッサエレメントの切替え時に、切替え前の被選択プロセッサエレメントから切替え後の被選択プロセッサエレメントへ受け渡すべき情報を、切替え前の被選択プロセッサエレメントから記憶部に格納し、この情報の格納後に、切替え前の被選択プロセッサエレメントを停止させるとともに、記憶部に格納された情報を用いて、切替え後の被選択プロセッサエレメントを動作させることにより、情報を切替え後の被選択プロセッサエレメントに対して容易且つ確実に受け渡すことができ、被選択プロセッサエレメントの切替えを容易且つ確実に行なうことができる（請求項1～請求項5）。

【0110】

(2) 切替え後の被選択プロセッサエレメントを停止させることにより、システムにおける消費電力を低減することができる他、プログラムが一続きになるように既述することができ、プログラミングやメンテナンス性が向上する（請求項1～請求項5）。

(3) プロセッサエレメントに対して要求された動作がこのプロセッサエレメント以外のプロセッサエレメントが処理すべきものである場合に、プロセッサエレメントの切替要求信号を出力することにより、プロセッサエレメントの切替えを容易且つ確実に行なうことができる。

【0111】

(4) 切替要求信号として切替制御用割込信号を用いることにより、切替要求信号を他の割込信号と一元的に管理することができる。

(5) 外部から信号を受信すると、被選択プロセッサエレメントに対してプロセッサエレメント用割込信号を出力することにより、被選択プロセッサエレメントを確実に制御することができる。

【0112】

(6) 2以上のプロセッサエレメントの内、処理の実行を許可されている実行許可プロセッサエレメントを指示するテーブルをそなえ、このテーブルによって指示される実行許可プロセッサエレメントに対して、プロセッサエレメント用割込信号を出力することにより、被選択プロセッサエレメントの切替制御を容易に

行なうことができる。

【0113】

(7) 2以上のプロセッサエレメントの内の一のプロセッサエレメントを優先的に動作させ、必要に応じて、一のプロセッサエレメントに代えて他のプロセッサエレメントを動作させることにより、優先的に動作させるプロセッサエレメントにおいて、他のプロセッサエレメントの動作を把握することができ、デバッグ等のメンテナンスやプログラミングが容易になり利便性が向上する。

【0114】

(8) 2以上のプロセッサエレメントが、互いに異なる機能を有することにより、各プロセッサエレメントに好適な処理を行なわせることができ、処理効率を向上させることができる。

(9) 特に、2以上のプロセッサエレメントとして、MPUとDSPとをそなえて構成することにより、MPUによってOSによる処理を行なわせるとともに、音声や画像、動画の処理や入出力処理等の、リアルタイム性が求められる処理をDSPによって行なわせることができる。

【0115】

(10) 外部から信号を受信すると、この信号を処理すべき一のプロセッサエレメントを2以上のプロセッサエレメントの内から選択して動作させることにより、処理を最適なプロセッサエレメントに行なわせることができ、処理効率を向上させることができる。

(11) 切替え機能が無効化して2以上のプロセッサエレメントを同時に動作させうる無効化部をそなえることにより、全てのプロセッサエレメントを同時に動作させることができ、例えば、高い処理性能が要求される処理を行なう場合に高速に処理を行なうことができ、更に、本発明のマルチプロセッサシステムを柔軟に使用することができる。

【図面の簡単な説明】

【図1】

本発明の第1実施形態としてのマルチプロセッサシステムの機能構成を示すブロック図である。

【図 2】

本発明の第 1 実施形態としてのマルチプロセッサシステムにおける機能構成および信号の流れを説明するためのブロック図である。

【図 3】

本発明の第 1 実施形態としてのマルチプロセッサシステムにおける動作許可テーブルの例を示す図である。

【図 4】

本発明の第 1 実施形態としてのマルチプロセッサシステムにおける割込管理テーブルの例を示す図である。

【図 5】

本発明の第 1 実施形態としてのマルチプロセッサシステムにおける、割り込み処理を除く通常の処理手順を示すシーケンス図である。

【図 6】

本発明の第 1 実施形態としてのマルチプロセッサシステムにおけるライブラリ動作を説明するための図である。

【図 7】

本発明の第 1 実施形態としてのマルチプロセッサシステムにおける割り込みが発生した場合の制御手法を説明するためのフローチャートである。

【図 8】

本発明の第 2 実施形態としてのマルチプロセッサシステムの割込管理テーブルの例を示す図である。

【図 9】

本発明の第 2 実施形態としてのマルチプロセッサシステムにおける割り込みが発生した場合の制御手法を説明するためのフローチャートである。

【図 1 0】

本発明の第 3 実施形態としてのマルチプロセッサシステムの割込管理テーブルの例を示す図である。

【図 1 1】

本発明の第 3 実施形態としてのマルチプロセッサシステムにおける DSP が処



理すべき割込処理（微少処理）が入った場合の制御手法を説明するためのフローチャートである。

【図 1 2】

本発明の第 4 実施形態としてのマルチプロセッサシステムの機能構成を示すブロック図である。

【図 1 3】

本発明の第 4 実施形態としてのマルチプロセッサシステムにおける機能構成および信号の流れを説明するためのブロック図である。

【符号の説明】

1 a, 1 b マルチプロセッサシステム

2, 2 a, 2 b PE（プロセッサエレメント，被選択プロセッサエレメント）

3 PE制御部（制御部）

4 共有メモリ部（記憶部）

3 1 割込管理部

3 2 動作許可テーブル（テーブル）

3 3 a, 3 3 b, 3 3 c 割込管理テーブル

3 4 PE実行許可部

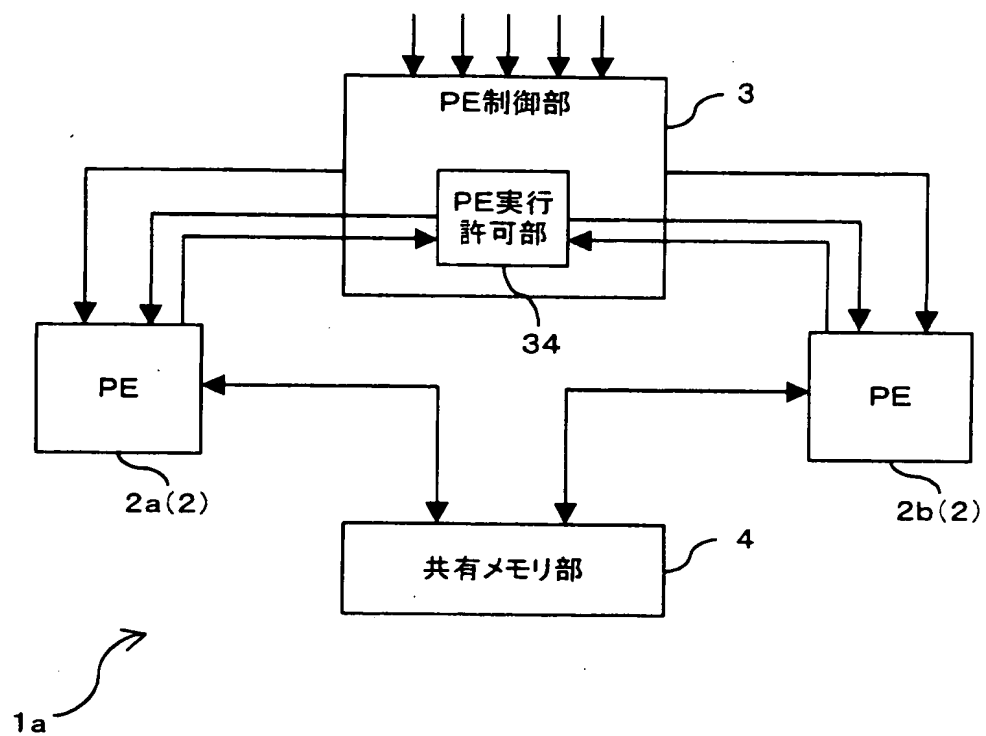
D 1 関数

L 1 ライブラリ

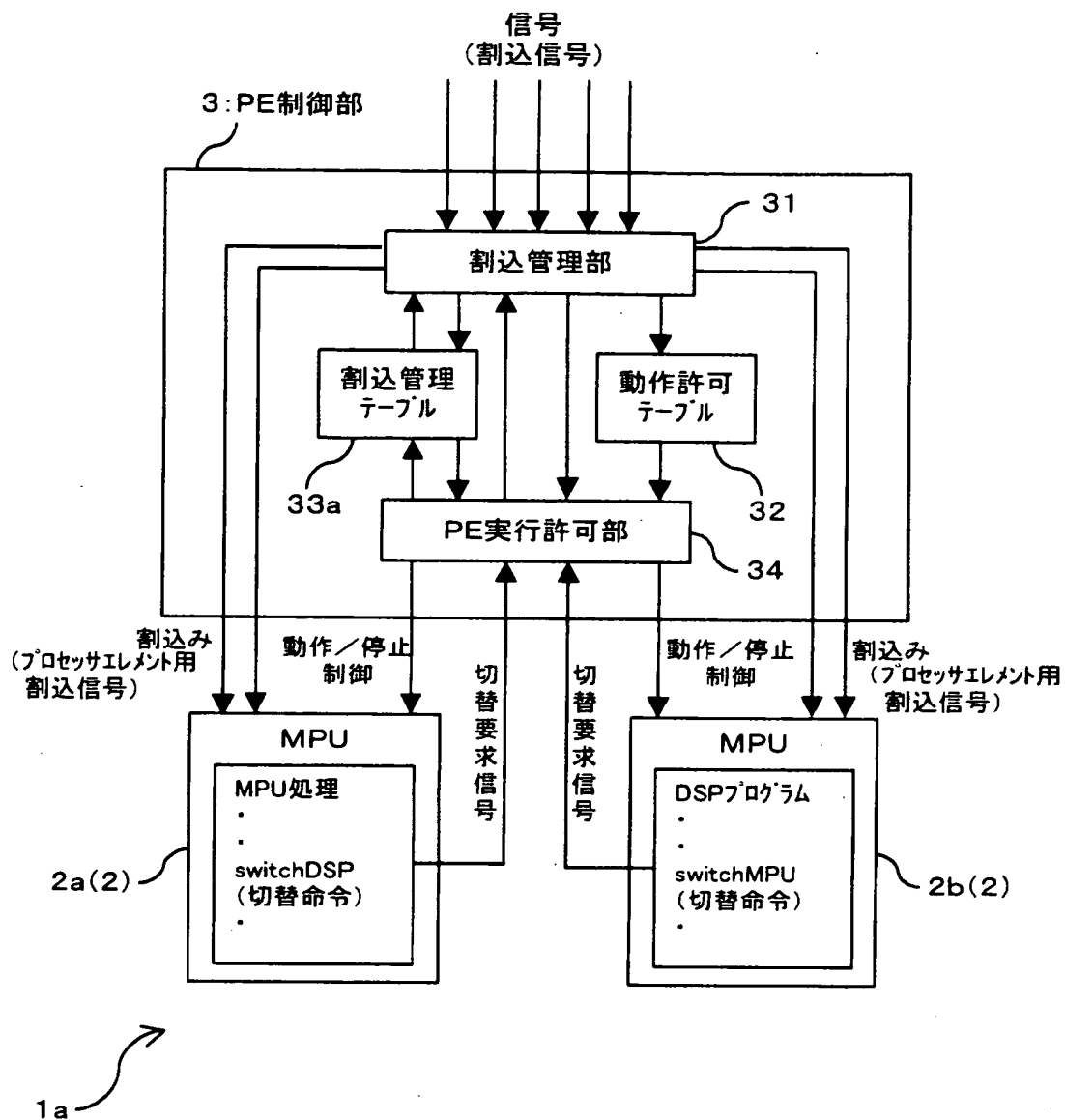
P 1 ユーザプログラム

【書類名】 図面

【図 1】




【図2】



【図3】


32



PE番号	実行許可フラグ
1 (PE2a)	0
2 (PE2b)	1
⋮	⋮
N	0

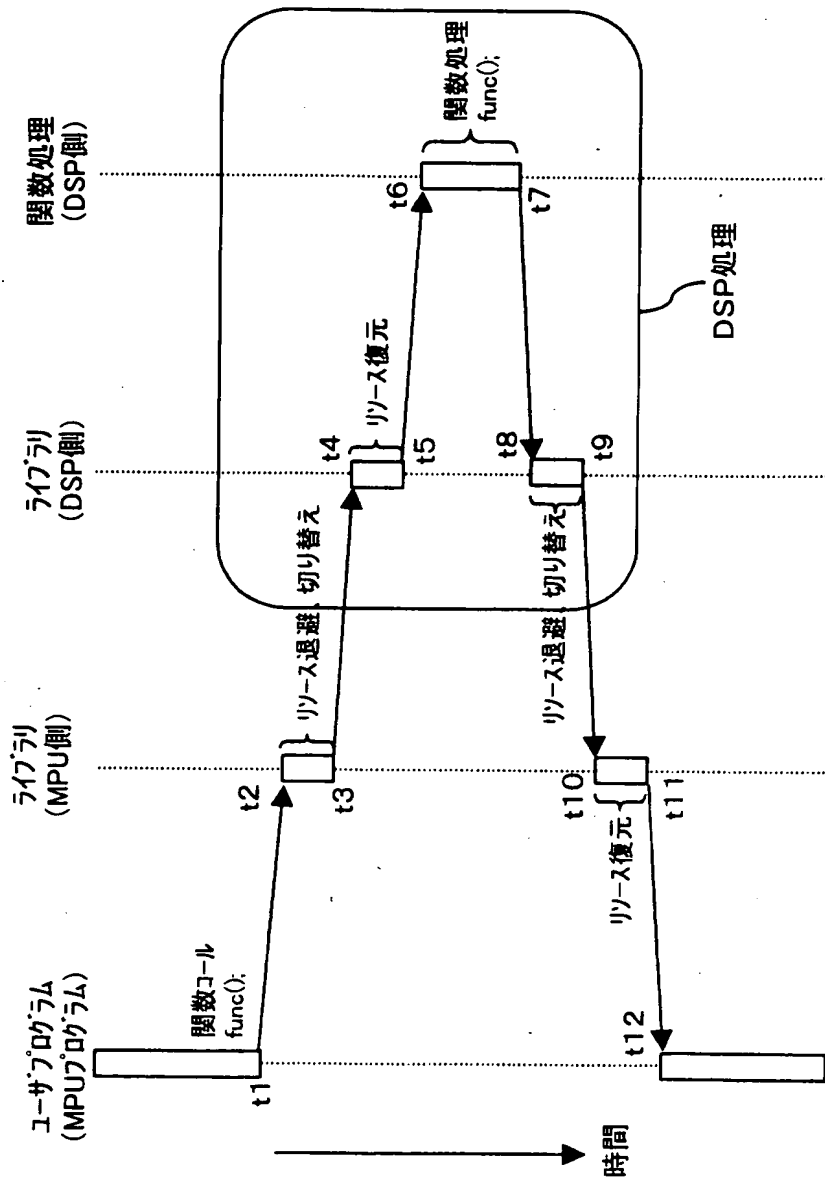
【図4】

33a

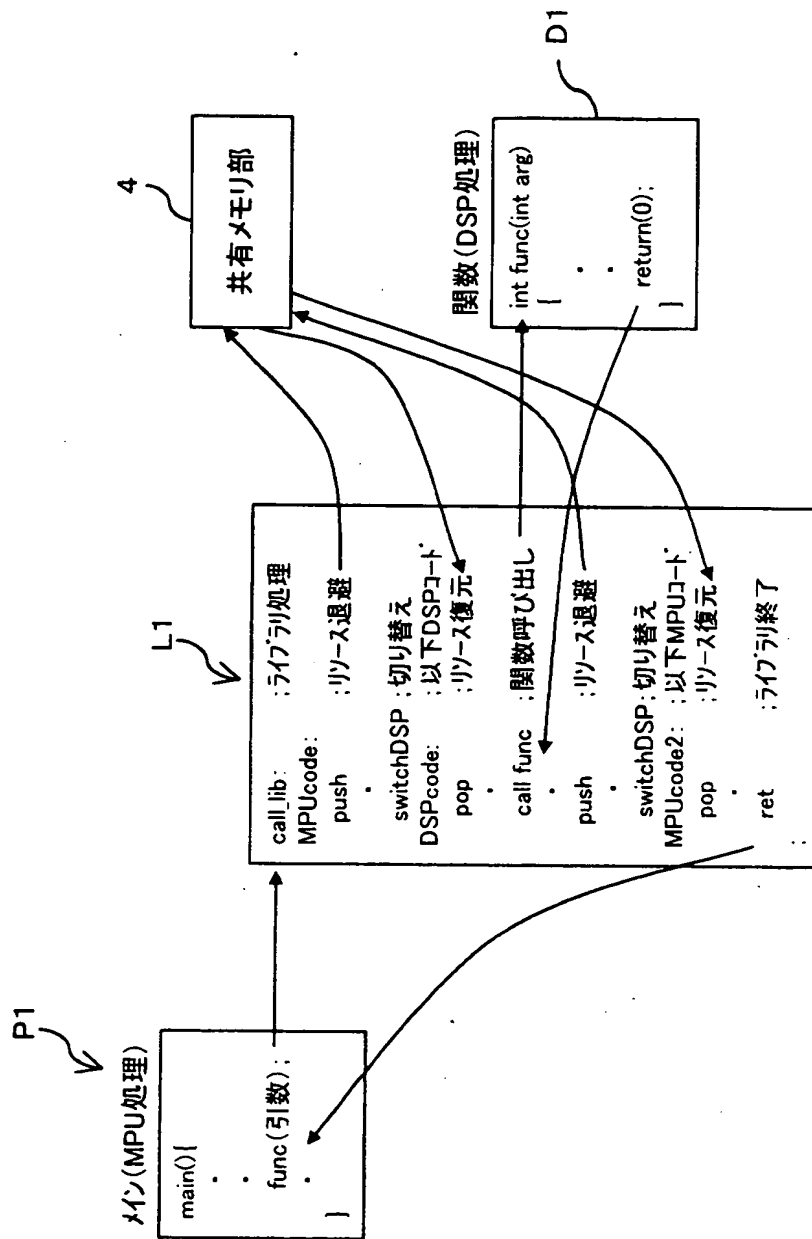


割込番号	割込対象	ペクタ番号	優先順位
1	ACTIVE	1	1
2	ACTIVE	2	N
⋮	⋮	⋮	⋮
N	ACTIVE	M	2

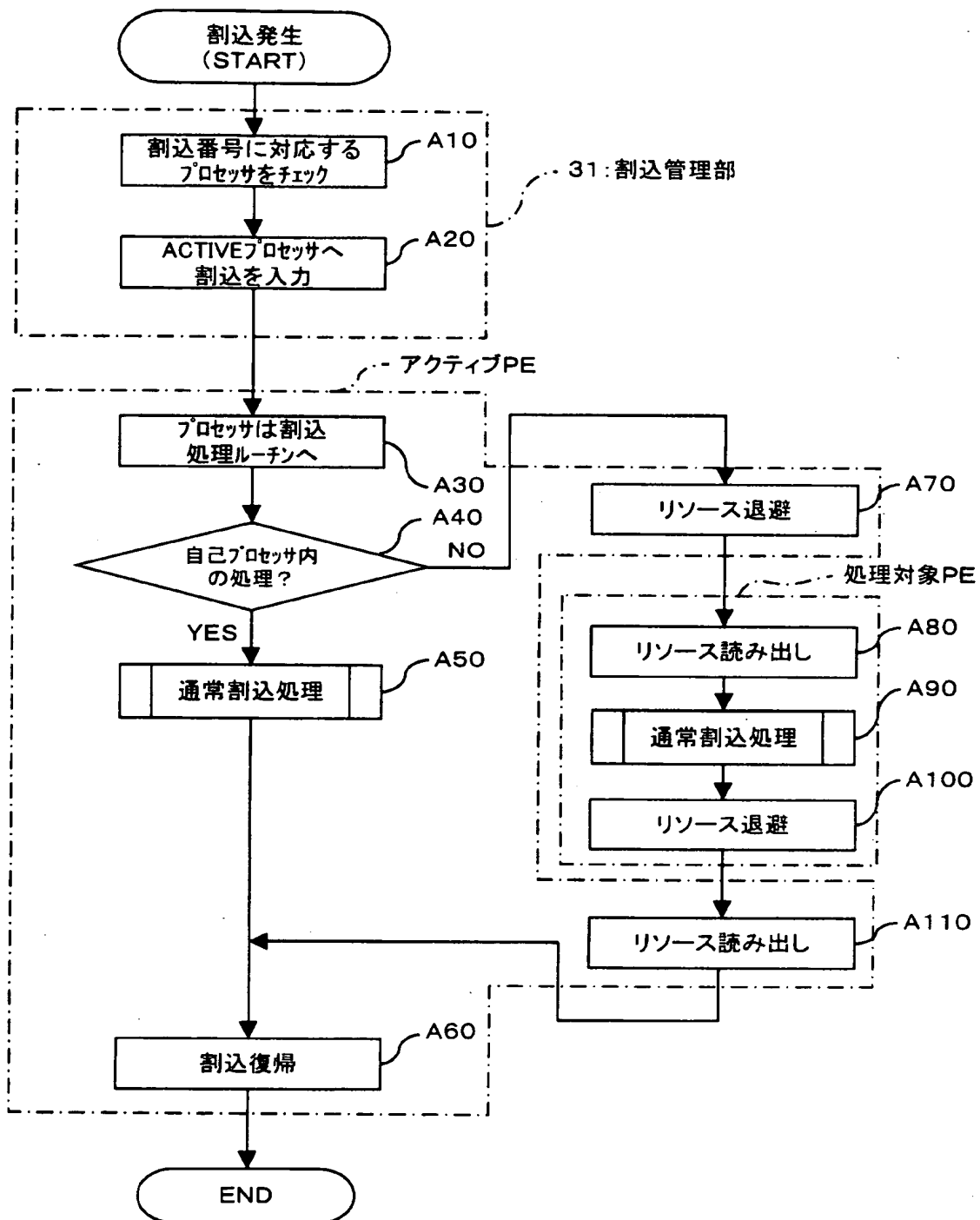
【図5】



【図 6】



【図 7】



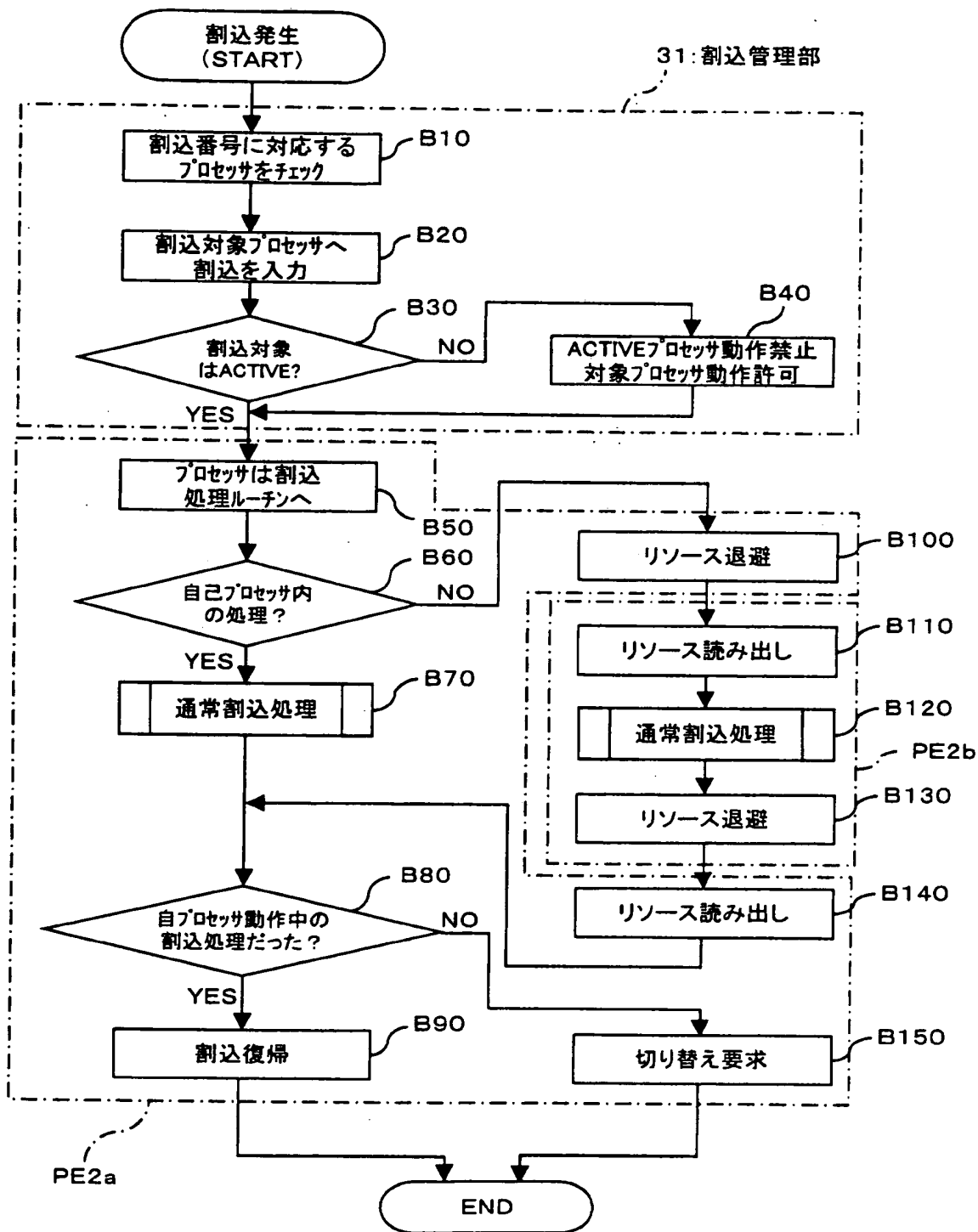
【図 8】

33b

割込番号	割込対象	ベクタ番号	優先順位
1	PE2a	1	1
2	PE2a	1	N
⋮	⋮	⋮	⋮
N	PE2a	M	2



【図 9】

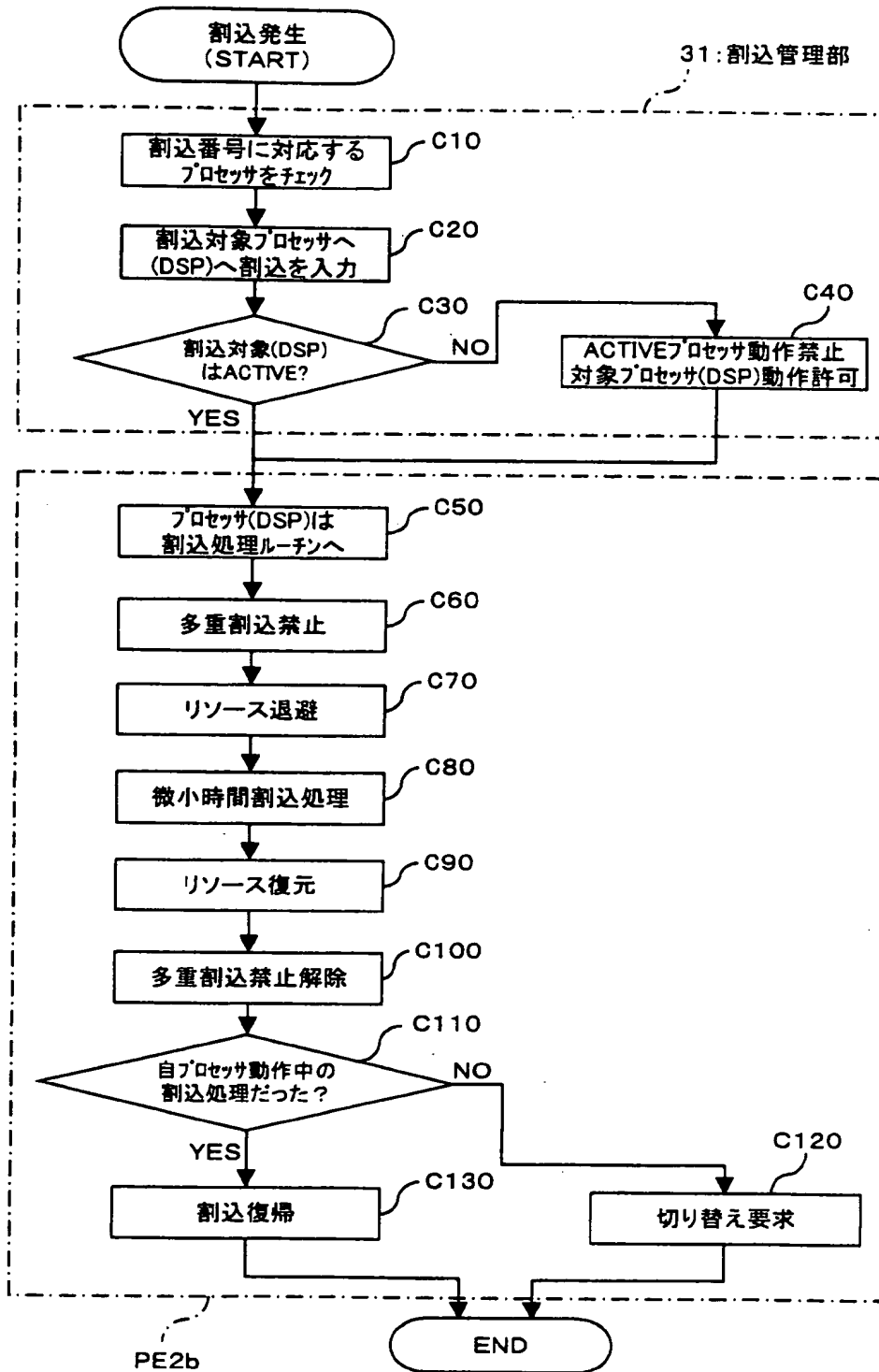


【図 1 0】

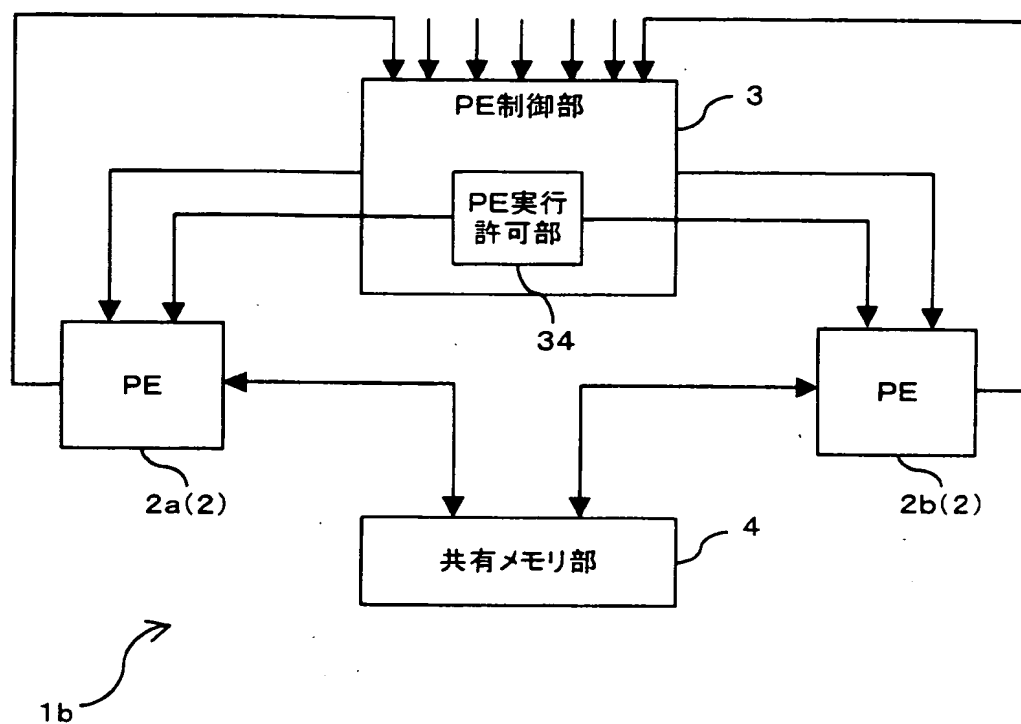
33c

割込番号	割込対象	ベクタ番号	優先順位
1	PE2b	1	1
2	PE2a	1	N
⋮	⋮	⋮	⋮
N	PE2a	M	2

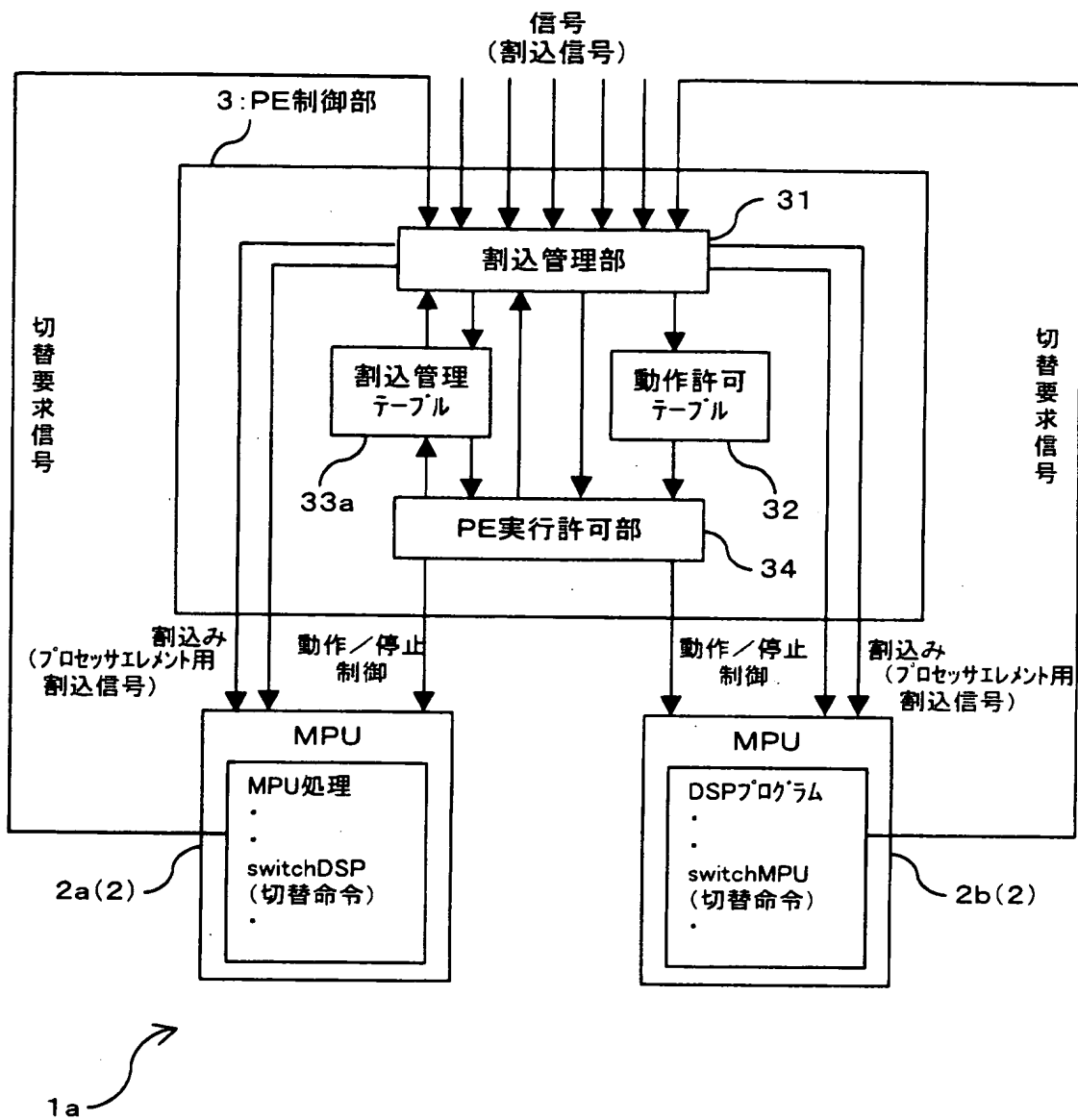
【図 11】



【図 1 2】



【図 13】



【書類名】            要約書

【要約】

【課題】    2以上のプロセッサエレメントの各機能を最大限に生かすことができるとともに、これらのプロセッサエレメント間における情報の受け渡しを確実に行なうことができ、更に、消費電力を低減することができるようにする。

【解決手段】    2以上のプロセッサエレメント2と、これらの2以上のプロセッサエレメント2の内から、一つのプロセッサエレメント2を被選択プロセッサエレメント2として選択して動作させる制御部3と、この制御部3による被選択プロセッサエレメント2の切替時に、切替前の被選択プロセッサエレメント2から切替後の被選択プロセッサエレメント2へ受け渡すべき情報を格納する記憶部4とをそなえるように構成する。

【選択図】            図1

出 願 人 履 歴 情 報

識別番号 [000005223]

1. 変更年月日 1996年 3月26日

[変更理由] 住所変更

住 所 神奈川県川崎市中原区上小田中4丁目1番1号

氏 名 富士通株式会社